PATENT **ATTORNEY DOCKET NO.: D6570-00003**

II. Claims

Claims 1-48 are pending in the present application. Claims 23-48 are withdrawn from consideration. No amendments are made to the claims, but the claims as originally presented are set forth in the listing below for the Examiner's convenience.

1. (original) A method of programming a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing a process through output signals at said outputs in response to input signals at said inputs, comprising the steps of:

displaying to a user on a monitor a graphical data entry user interface for a plurality of sequential steps, said graphical data entry user interface representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and respective outputs to be initiated by said programmable logic controller at respective ones of said sequential steps;

receiving, via said graphical data entry user interface, an identification of at least one input selected by said user to be monitored for at least one of said sequential steps and an identification of at least one output selected by said user to be initiated for said at least one of said sequential steps;

converting said identification of said at least one input selected by said user into an input control data table, said input control data table including a plurality of input control data elements, each of said input control data elements corresponding to a respective one of said plurality of sequential steps, a respective one of said input control data elements representing said at least one input selected by said user; and

converting said identification of said at least one output selected by said user into an output data table, said output data table including a plurality of output data elements, each of said output data elements corresponding to a respective one of said plurality of sequential steps, a

respective one of said output data elements representing said at least one output selected by said user.

2. (original) The method of claim 1, wherein said graphical data entry user interface includes a timer enable command option for each of said plurality of sequential steps and a timer value option for each of said plurality of sequential steps.

3. (original) The method of claim 2, further comprising the steps of:

receiving, via said graphical data entry user interface, a selection by said user of a timer enable command for at least one of said plurality of sequential steps;

receiving, via said graphical data entry user interface, a selection by said user of a timer value for said one of said plurality of sequential steps; and

creating a timer value data table including at least one timer value data element, said timer value data element representing said timer value,

wherein a respective one of said input control data elements represents said timer enable command for said one of said sequential steps.

4. (original) The method of claim 3, wherein said input control data element includes a plurality of bits, a subset of said plurality of bits representing individual inputs of said programmable logic controller and at least a remaining one of said plurality of bits representing said timer enable command.

5. (original) The method of claim 1, wherein said input control data element includes a plurality of bits and a subset of said plurality of bits represents individual inputs of said programmable logic controller.

6. (original) The method of claim 1, wherein said graphical data entry user interface is a check grid.

7. (original) The method of claim 1, further comprising the step of downloading said input control data table and said output data table to said programmable logic controller.

8. (original) The method of claim 1, wherein said output data element includes a plurality of bits and a subset of said plurality of bits represents individual outputs of said programmable logic controller.

9. (original) An apparatus for programming a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing a process through output signals at said outputs in response to input signals at said inputs, comprising:

means for displaying to a user on a monitor a graphical data entry user interface for a plurality of sequential steps, said graphical data entry user interface representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and

respective outputs to be initiated by said programmable logic controller at respective ones of said sequential steps;

means for receiving, via said graphical data entry user interface, an identification of at least one input selected by said user to be monitored for at least one of said sequential steps and an identification of at least one output selected by said user to be initiated for said at least one of said sequential steps;

means for converting said identification of said at least one input selected by said user into an input control data table, said input control data table including a plurality of input control data elements, each of said input control data elements corresponding to a respective one of said plurality of sequential steps, a respective one of said input control data elements representing said at least one input selected by said user; and

means for converting said identification of said at least one output selected by said user into an output data table, said output data table including a plurality of output data elements, each of said output data elements corresponding to a respective one of said plurality of sequential steps, a respective one of said output data elements representing said at least one output selected by said user.

10. (original) The apparatus of claim 9, wherein said graphical data entry user interface includes a timer enable command option for each of said plurality of sequential steps and a timer value option for each of said plurality of sequential steps.

11. (original) The apparatus of claim 10, further comprising:

means for receiving, via said graphical data entry user interface, a selection by said user of a timer enable command for at least one of said plurality of sequential steps;

means for receiving, via said graphical data entry user interface, a selection by said user of a timer value for said one of said plurality of sequential steps; and

means for creating a timer value data table including at least one timer value data element, said timer value data element representing said timer value,

wherein a respective one of said input control data elements represents said timer enable command for said one of said sequential steps.

12. (original) The apparatus of claim 11, wherein said input control data element includes a plurality of bits, a subset of said plurality of bits representing individual inputs of said programmable logic controller and at least a remaining one of said plurality of bits representing said timer enable command.

13. (original) The apparatus of claim 9, wherein said input control data element includes a plurality of bits and a subset of said plurality of bits represents individual inputs of said programmable logic controller.

14. (original) The apparatus of claim 9, wherein said graphical data entry user interface is a check grid displayed on said monitor.

15. (original) The apparatus of claim 9, further comprising means for downloading said input control data table and said output data table to said programmable logic controller.

16. (original) The apparatus of claim 9, wherein said output data element includes a plurality of bits and a subset of said plurality of bits represents individual outputs of said programmable logic controller.

17. (original) A computer-readable medium encoded with a computer program code for programming a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing a process through output signals at said outputs in response to input signals at said inputs, the medium comprising:

a first code segment for displaying to a user on a monitor a graphical data entry user interface for a plurality of sequential steps, said graphical data entry user interface representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and respective outputs to be initiated by said programmable logic controller at respective ones of said sequential steps;

a second code segment for receiving, via said graphical data entry user interface, an identification of at least one input selected by said user to be monitored for at least one of said sequential steps and an identification of at least one output selected by said user to be initiated for said at least one of said sequential steps;

a third code segment for converting said identification of said at least one input selected by said user into an input control data table, said input control data table including a plurality of input control data elements, each of said input control data elements corresponding to a

respective one of said plurality of sequential steps, a respective one of said input control data elements representing said at least one input selected by said user; and

a fourth code segment for converting said identification of said at least one output selected by said user into an output data table, said output data table including a plurality of output data elements, each of said output data elements corresponding to a respective one of said plurality of sequential steps, a respective one of said output data elements representing said at least one output selected by said user.

18. (original) The computer-readable medium of claim 17, wherein said graphical data entry user interface includes a timer enable command option for each of said plurality of sequential steps and a timer value option for each of said plurality sequential steps.

19. (original) The computer-readable medium of claim 18, further comprising:

a fifth code segment for receiving, via said graphical data entry user interface, a selection by said user of a timer enable command for at least one of said plurality of sequential steps;

a sixth code segment for receiving, via said graphical data entry user interface, a selection by said user of a timer value for said one of said plurality of sequential steps; and

a seventh code segment for creating a timer value data table including at least one timer value data element, said timer value data element representing said timer value,

wherein a respective one of said input control data elements represents said timer enable command for said one of said sequential steps.

20. (original) A computer data signal embodied in a carrier wave encoded with computer program code for programming a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing a process through output signals at said outputs in response to input signals at said inputs, comprising:

a first code segment for displaying to a user on a monitor a graphical data entry user interface for a plurality of sequential steps, said graphical data entry user interface representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and respective outputs to be initiated by said programmable logic controller at respective ones of said sequential steps;

a second code segment for receiving, via said graphical data entry user interface, an identification of at least one input selected by said user to be monitored for at least one of said sequential steps and an identification of at least one output selected by said user to be initiated for said at least one of said sequential steps;

a third code segment for converting said identification of said at least one input selected by said user into an input control data table, said input control data table including a plurality of input control data elements, each of said input control data elements corresponding to a respective one of said plurality of sequential steps, a respective one of said input control data elements representing said at least one input selected by said user; and

a fourth code segment for converting said identification of said at least one output selected by said user into an output data table, said output data table including a plurality of output data elements, each of said output data elements corresponding to a respective one of said plurality of sequential steps, a respective one of said output data elements representing said at least one output selected by said user.

21. (original) The computer data signal of claim 20, wherein said graphical data entry user interface includes a timer enable command option for each of said plurality of sequential steps and a timer value option for each of said sequential steps.

22. (original) The computer data signal of claim 21, further comprising:

a fifth code segment for receiving, via said graphical data entry user interface, a selection by said user of a timer enable command for at least one of said plurality of sequential steps;

a sixth code segment for receiving, via said graphical data entry user interface, a selection by said user of a timer value for said one of said plurality of sequential steps; and

a seventh code segment for creating a timer value data table including at least one timer value data element, said timer value data element representing said timer value,

wherein a respective one of said input control data elements represents said timer enable command for said one of said sequential steps.

23. (withdrawn) A method of controlling a process with a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing said process through signals at said outputs in response to input signals at said inputs, comprising the steps of:

accessing with said programmable logic controller an input control data element for a sequential step and an output data element for said sequential step from an input control data table and an output data table, respectively, said input control data table including input control data elements for a plurality of sequential steps that include said sequential step and said output data table including a plurality of output data elements for said plurality of sequential steps;

providing output signals at outputs of said programmable logic controller identified by said output data element to be activated for said sequential step;

monitoring inputs identified by said input control data element to be monitored for said sequential step; and

performing a next one of said plurality of sequential steps if an input signal is detected for at least one of said monitored inputs.

24. (withdrawn) The method of claim 23, wherein at least one input control data element represents a timer enable command for a respective one of said plurality of sequential steps, said method further comprising the steps of:

accessing with said programmable logic controller a timer value from a timer value data table for said sequential step;

enabling a timer for a time period indicated by said timer value for said sequential step; and

performing a next sequential step when said time period expires.

25. (withdrawn) The method of claim 24, wherein said input control data element includes a plurality of bits and a subset of said plurality of bits represents individual inputs of said programmable logic controller and at least one of said subset of bits represents said timer enable command.

26. (withdrawn) The method of claim 23, wherein said input control data element includes a plurality of bits and a subset of said plurality of bits represents individual inputs of said programmable logic controller.

27. (withdrawn) The method of claim 23, wherein said input control data element and said output data element are accessed from a local storage medium.

28. (withdrawn) The method of claim 23, wherein said output data element includes a plurality of bits and a subset of said plurality of bits represents individual outputs of said programmable logic controller.

29. (withdrawn) The method of claim 23, further comprising the steps of:

creating said input control data table; and

creating said output data table.

30. (withdrawn) The method of claim 29, wherein said steps of creating said input control data table and said output data table include the following steps:

displaying to a user on a monitor a graphical data entry user interface for a plurality of sequential steps, said graphical data entry user interface representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and respective

outputs to be initiated by said programmable logic controller at respective ones of said sequential steps;

receiving, via said graphical data entry user interface, an identification of at least one input selected by said user to be monitored for at least one of said sequential steps and an identification of at least one output selected by said user to be initiated for said at least one of said sequential steps;

converting said identification of said at least one input selected by said user into said input control data table; and

converting said identification of said at least one output selected by said user into said output data table.

31. (withdrawn) The method of claim 30, wherein said graphical data entry user interface includes a timer enable command option for each of said plurality of sequential steps and a timer value option for each of said plurality of sequential steps.

32. (withdrawn) The method of claim 31, further comprising the steps of:

receiving, via said graphical data entry user interface, a selection by said user of a timer enable command for at least one of said plurality of sequential steps;

receiving, via said graphical data entry user interface, a selection by said user of a timer value for said one of said plurality of sequential steps; and

creating a timer value data table including at least one timer value data element, said timer value data element representing said timer value.

33. (withdrawn) The method of claims 32, wherein at least one input control data element represents a timer enable command for a respective one of said plurality of sequential steps, said method further comprising the steps of:

accessing with said programmable logic controller a timer value from a timer value data table for said sequential step;

enabling a timer for a time period indicated by said timer value for said sequential step; and

performing a next sequential step when said time period expires.

34. (withdrawn) A programmable logic controller for controlling a process, said programmable logic controller comprising:

a plurality of inputs and a plurality of outputs, said programmable logic controller directing said process via signals provided at said outputs in response to input signals at said inputs;

means for accessing an input control data element for a sequential step and an output data element for said sequential step from an input control data table and an output data table, respectively, said input control data table including input control data elements for a plurality of sequential steps that include said sequential step and said output data table including a plurality of output data elements for said plurality of sequential steps;

means for providing output signals at outputs of said programmable logic controller identified by said output data element to be activated for said sequential step;

means for monitoring inputs identified by said input control data element to be monitored for said sequential step; and

means for performing a next one of said plurality of sequential steps if an input signal is detected for at least one of said monitored inputs.

35. (withdrawn) The programmable logic controller of claim 34, wherein at least one input control data element represents a timer enable command for a respective one of said plurality of sequential steps, said programmable logic controller further comprising:

means for accessing a timer value from a timer value data table for said sequential step;

means for enabling a timer for a time period indicated by said timer value for said sequential step; and

means for performing a next sequential step when said time period expires.

36. (withdrawn) The programmable logic controller of claim 35, wherein said input control data element includes a plurality of bits and a subset of said plurality of bits represents individual inputs of said programmable logic controller and at least one of said subset of bits represents said timer enable command.

37. (withdrawn) The programmable logic controller of claim 34, wherein said input control data element includes a plurality of bits and a subset of said plurality of bits represents individual inputs of said programmable logic controller.

38. (withdrawn) The programmable logic controller of claim 34, wherein said input control data element and said output data element are accessed from a local storage medium.

39. (withdrawn) The programmable logic controller of claim 34, wherein said output data element includes a plurality of bits and a subset of said plurality of bits represents individual outputs of said programmable logic controller.

40. (withdrawn) The programmable logic controller of claim 34, further comprising:

means for creating said input control data table; and

means for creating said output data table.

41. (withdrawn) The programmable logic controller of claim 40, wherein said means for creating said input control data table and said output data table include the following:

means for displaying to a user on a monitor a graphical data entry user interface for a plurality of sequential steps, said graphical data entry user interface representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and respective outputs to be initiated by said programmable logic controller at respective ones of said sequential steps;

means for receiving, via said graphical data entry user interface, an identification of at least one input selected by said user to be monitored for at least one of said sequential steps and

an identification of at least one output selected by said user to be initiated for said at least one of said sequential steps;

means for converting said identification of said at least one input selected by said user into said input control data table; and

means for converting said identification of said at least one output selected by said user into said output data table.

42. (withdrawn) The programmable logic controller of claim 41, wherein said graphical data entry user interface includes a timer enable command option for each of said plurality of sequential steps and a timer value option for each of said plurality of sequential steps.

43. (withdrawn) The programmable logic controller of claim 42, further comprising:

means for receiving, via said graphical data entry user interface, a selection by said user of a timer enable command for at least one of said plurality of sequential steps;

means for receiving, via said graphical data entry user interface, a selection by said user of a timer value for said one of said plurality of sequential steps; and

means for creating a timer value data table including at least one timer value data element, said timer value data element representing said timer value.

44. (withdrawn) The programmable logic controller of claims 43, wherein at least one input control data element represents a timer enable command for a respective one of said plurality of sequential steps, further comprising:

means for accessing with said programmable logic controller a timer value from a timer value data table for said sequential step;

means for enabling a timer for a time period indicated by said timer value for said sequential step; and

means for performing a next sequential step when said time period expires.

45. (withdrawn) A computer-readable medium encoded with a computer program code for controlling a process with a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing said process through signals at said outputs in response to input signals at said inputs, the medium comprising:

a first code segment that causes said programmable logic controller to access an input control data element for a sequential step and an output data element for said sequential step from an input control data table and an output data table, respectively, said input control data table including input control data elements for a plurality of sequential steps that include said sequential step and said output data table including a plurality of output data elements for said plurality of sequential steps;

a second code segment that causes said programmable logic controller to provide output signals at outputs of said programmable logic controller identified by said output data element to be activated for said sequential step;

a third code segment that causes said programmable logic controller to monitor inputs identified by said input control data element to be monitored for said sequential step; and

a fourth code segment that causes said programmable logic controller to perform a next one of said plurality of sequential steps if an input signal is detected for at least one of said monitored inputs.

46. (withdrawn) The computer-readable medium of claim 45, wherein at least one input control data element represents a timer enable command for a respective one of said plurality of sequential steps, said computer-readable medium further comprising:

a fifth code segment that causes said programmable logic controller to access a timer value from a timer value data table for said sequential step;

a sixth code segment that causes said programmable logic controller to enable a timer for a time period indicated by said timer value for said sequential step; and

a seventh code segment that causes said programmable logic controller to perform a next sequential step when said time period expires.

47. (withdrawn) A computer data signal embodied in a carrier wave encoded with computer program code for controlling a process with a programmable logic controller, said programmable logic controller including a plurality of inputs and a plurality of outputs, said programmable logic controller directing a process through output signals at said outputs in response to input signals at said inputs, said computer data signal comprising:

a first code segment that causes said programmable logic controller to access an input control data element for a sequential step and an output data element for said sequential step

from an input control data table and an output data table, respectively, said input control data table including input control data elements for a plurality of sequential steps that include said sequential step and said output data table including a plurality of output data elements for said plurality of sequential steps;

a second code segment that causes said programmable logic controller to provide output signals at outputs of said programmable logic controller identified by said output data element to be activated for said sequential step;

a third code segment that causes said programmable logic controller to monitor inputs identified by said input control data element to be monitored for said sequential step; and

a fourth code segment that causes said programmable logic controller to perform a next one of said plurality of sequential steps if an input signal is detected for at least one of said monitored inputs.

48. (withdrawn) The computer data signal of claim 47, wherein at least one input control data element represents a timer enable command for a respective one of said plurality of sequential steps, said computer data signal further comprising:

a fifth code segment that causes said programmable logic controller to access a timer value from a timer value data table for said sequential step;

a sixth code segment that causes said programmable logic controller to enable a timer for a time period indicated by said timer value for said sequential step; and

a seventh code segment that causes said programmable logic controller to perform a next sequential step when said time period expires.

PATENT **ATTORNEY DOCKET NO.: D6570-00003**

III. Remarks

Applicant is grateful to Examiner Nguyen and Examiner Kincaid for their time in the telephone interview of July 8, 2004 and their agreement to withdrawal the July 2, 2004 Final Office Action and issue this new non-final Office Action.

Reconsideration and withdrawal of the new rejections are respectfully requested in view of the following arguments.

A. Rejection under 35 U.S.C. § 102

The Action rejects Claims 1, 5, 8, 9, 16, 17 and 20 as being anticipated by U.S. Patent No. 5,867,382 to McLaughlin. The Applicant has carefully considered this rejection and respectfully submits that the Examiner has misconstrued McLaughlin, as McLaughlin does not teach a programmable logic controller (PLC) programming method or apparatus. Reconsideration and withdrawal of this rejection are respectfully requested in view of the following arguments.

Claim 1 is directed to a method of **programming** a programmable logic controller (PLC). The PLC includes a plurality of inputs and outputs and directs a process through output signals at the outputs in response to input signals at the inputs. The claimed method recites, among other things, four steps for **programming** the PLC. As recognized by McLaughlin, programs, such as program 204 of McLaughlin (FIGS. 2, 4, 24, 26) are typically written in a programming language called "ladder logic." (Column 7, Lines 24) The processor executes an I/O scan where input status data are read from the I/O interfaces connected to the PLC and output data are written to the I/O interfaces from the PLC in accordance with the particular rung of the ladder logic program being executed.

The Applicant has realized that there are difficulties in debugging ladder logic programs and in reprogramming ladder logic programs to perform different sequences of operations. Indeed, skilled programmers are required for these programming tasks. Based on this

realization, the Applicant developed an intuitive graphical user interface that allows a person without any ladder logic knowledge to program and debug the PLC.

The claimed method is best understood with reference to FIGS. 2 and 3 of the application. FIG. 2 shows a graphical user interface that allows a programmer to select inputs and outputs for the PLC for a series of sequential steps to be preformed in a process, such as an assembly process. The selections are converted to an input control table and an output control data table that can then be provided to a PLC, which performs the sequential steps according to the programmed data in the control tables. This methodology is essentially embodied in the four programming steps of Claim 1.

McLaughlin provides an interface between the PLC program 2604 and the I/O modules 2608, 2612. (FIG. 26). Currently, if an I/O module is changed to a newer version, or different manufacturer, the PLC program 2604 will have to me changed to accommodate the new I/O module. A PLC uses a memory address scheme that is dependent on the manufacturer, style and revision of the card. McLaughlin provides a methodology where the ladder logic program 2604 calls the virtual I/O rack program 2602 to get I/O assignments. The virtual rack 2602 will then give the program 2604 the status of the I/O.

Most importantly, McLaughlin provides nothing, other than conventional ladder logic programming methods, directed to the actual programming of the PLC operating program, i.e., program 2602. Specifically, and referring to FIG. 2 of McLaughlin, McLaughlin provides that the ladder logic program 204 is developed on programming terminal 108 (i.e., a PC) and downloaded to memory 202 of the PLC 102. McLaughlin does not provide for a new method of writing the control program 204 of McLaughlin, but rather relies on conventional ladder logic programming.

Applicant has developed, and claims in Claim 1, a new method of programming a PLC to direct a process. This process uses a configurable graphical user interface that graphically represents the ladder logic program to the user. This alleviates the need for the programmer to

know ladder logic, as the inputs and outputs selected by the user in Applicant's user friendly, intuitive interface are converted to input and output control data tables, which can be downloaded to the PLC for execution.

Conversely, McLaughlin still requires the user to program the main control program 2604 of the PLC in standard ladder logic. This program is then directly downloaded to the PLC. McLaughlin's disclosure then deals with configuration of I/O modules, not programming. Indeed, McLaughlin recognizes that the main advantage and purpose of the method of McLaughlin is that "the programmer only needs to program the logic of the control algorithms and does not need to program where the various I/O devices will be located or when the various I/O devices were installed in the generic control program. In other words, the Virtual Rack module 500 resembles a configuration language, rather than a process control system 100." (Column 7, Lines 52-59).

It must be stressed that McLaughlin does not provide a method of programming a programmable logic controller that directs a process through output signals in response to input signals. No graphical data entry user interface is displayed to the user, as recited in Claim 1, "representing respective inputs to be monitored by said programmable logic controller at each of said sequential steps and respective outputs to be initiated by said programmable logic controller at respective ones of said sequential steps." As described in more detail below in addressing the portions of McLaughlin relied upon by the Examiner, any graphical interface that is displayed to the user by McLaughlin deals with configuring the I/O connected to the PLC and not the individual inputs to be monitored and outputs to be triggered at sequential steps of the process being executed by the PLC. It follows that any graphical user interface of McLaughlin does not receive an identification of inputs selected by the user to be monitored at steps or outputs to be initiated at steps, nor converting these inputs and outputs to input control and output control data tables for use by the PLC.

In rejecting Claim 1, the Examiner cites to FIGS. 13-23, and specifically cites to FIG. 22 and Column 18, Lines 3-20, as providing the recited graphical user interface. However, as argued above, the graphical interface of McLaughlin is a configuration interface that enables the user to configure the I/O modules, not to program the PLC to perform the steps to direct a process. That step, i.e., programming the steps of the PLC program, is performed using conventional ladder logic programming, not a graphical user interface as claimed.

In describing the INPUT section 2202 and output section 2204 of FIG. 22, McLaughlin refers back to the description provided previously for, for example, FIGS. 16-18. Referring to FIG. 16, the FORCE configuration field of FIG. 16 allows the user to force on outputs to test field wiring and equipment *during the installation* of the virtual rack module 500. (Column 15, Lines 13-24). The interface of FIG. 16 in no way allows the user to program the sequential steps executed by the PLC, or associated inputs to be monitored and outputs to be initiated for each step based on the inputs. FIG. 17 shows an interface for configuring the ANALOG IN module type. The FACTOR configuration field allows for setting of an engineering scaling factor for each channel and has nothing to do with setting of inputs and outputs for each sequential step of a PLC run process. (Column 15, Line 62-Column 16, Line 11). Similarly, FIG. 18 allows for the configuration of the ANALOG OUT module type, specifically for specifying the minimum and maximum signal range for an analog channel. (Column 16, Lines 36-40).

In summary, the graphical interfaces of McLaughlin allow only for the configuration of the I/O modules, and do not allow for the actual programming of inputs to be monitored and outputs to be initiated for sequential steps of the PLC program. To the extent that McLaughlin provides any disclosure with respect to programming the sequential steps to be executed by the PLC, McLaughlin provides only that this may be done using conventional ladder logic programming methodologies. *See e.g.,* Column 5, Lines 57-65. It is the need for the use of, knowledge of and dependence on these conventional ladder logic programming methodologies that Applicant has sought to eliminate (or minimize) and that is claimed in Claim 1.

Therefore, it is submitted that Claim 1 is not anticipated by McLaughlin and is allowable. Reconsideration and withdrawal of this rejection are respectfully requested. Claims 2-8 depend from McLaughlin and, it is submitted, are also allowable.

Independent Claims 9 (directed to an apparatus for programming a PLC), 17 (directed to a computer-readable medium encoded with code for programming a PLC) and 20 (directed to a computer data signal in a carrier wave encoded with code for programming a PLC) parallel Claim 1 and, it is submitted, are allowable for at least the reasons set forth above. Claims 10-16, 18-19, and 21-22 depend from independent Claims 9, 17 and 20, respectively, and are also allowable. Reconsideration and withdrawal of this rejections are respectfully requested.

B. Claim Rejection under 35 U.S.C. § 103

The Action rejects Claims 2-4, 10-13, 18, 19, 21 and 22 as being unpatentable over McLaughlin in view of U.S. Patent No. 6,678,636 to Coleman et al. These claims depend from allowable independent Claims 9, 17 and 20. For at least the reasons set forth above, these claims are allowable over the cited references.